

## REMARKS

This paper is being provided in response to the Office Action dated May 22, 2007 for the above-referenced application. In this response, Applicant has amended Claims 1, 4, 8, 18, 20, 21, 23, 26, 30, 40, 42, and 43 in order to clarify that which Applicant deems to be the claimed invention. Applicant respectfully submits that the amendments to the claims do not add new matter.

The rejection of Claims 1-43 under 35 U.S.C. 102(e) as being anticipated by Fee et al. (U.S. Patent No. 6,516,393, hereinafter referred to as “Fee”) is hereby traversed and reconsideration thereof is respectfully requested.

Claim 1, as amended herein, recites a method for distributed lock management comprising: determining, by a requesting node requesting a lock, a current lock owner of the lock in accordance with an indicator specifying which one or more nodes are available as lock owners; and sending a lock request message to said current lock owner; and wherein said requesting node waits for an indication as to whether the lock request message has been granted or denied, and said determining is performed by said requesting node prior to said sending for each request, and wherein, each time said lock is granted, an expiration time is determined indicating that said lock expires automatically when said expiration time is reached. Claims 2-17 depend from Claim 1.

Claim 18, as amended herein, recites a method for distributed lock management in a data storage system comprising: determining, by a requesting node requesting a lock, a current lock owner of the lock in accordance with an indicator specifying which one or more nodes are

available as lock owners; and sending a lock request message to said current lock owner, and wherein said determining is performed by said requesting node prior to said sending for each request and wherein, each time said lock is granted, an automatic expiration period is determined indicating when a current grant of said lock at said each time automatically expires. Claims 19-21 depend from Claim 18.

Claim 22 recites a data storage system comprising: a plurality of processors which are endpoints in said data storage system; a message switch used by said plurality of processors to send transmissions, said transmissions including a lock request message and a lock response message; each of said plurality of processors including machine executable instructions on a computer readable medium for processing a lock request from said each processor to another processor that: determines, prior to sending a lock request message, a current lock owner in accordance with an indicator specifying which one or more of said plurality of processors are available as lock owners; and sending said lock request message to said current lock owner; and each of said plurality of processors including machine executable instructions on a computer readable medium for processing a received lock request message when said each processor is indicated as a lock owner by said indicator of another processor that sent said received lock request message that: determines whether a requested lock in accordance with said received lock request message is currently granted; granting to said requested lock to a requesting processor if said requested lock is not currently granted, and otherwise denying said requested lock to said requested node, wherein said lock owner determines an automatic expiration time in accordance with a requested lock time included in said received lock request message.

Claim 23, as amended herein, recites a computer program product for distributed lock management comprising: executable code that determines, by a requesting node requesting a lock, a current lock owner of the lock in accordance with an indicator specifying which one or more nodes are available as lock owners; and executable code that sends a lock request message to said current lock owner; and wherein said requesting node waits for an indication as to whether the lock request message has been granted or denied, and said executable code that determines is executed by said requesting node prior to executing said executable code that sends for each request, and wherein, each time said lock is granted, an expiration time is determined indicating that said lock expires automatically when said expiration time is reached.

Claims 24-39 depend from Claim 23.

Claim 40, as amended herein, recites a computer program product for distributed lock management in a data storage system comprising: executable code that determines, by a requesting node requesting a lock, a current lock owner of the lock in accordance with an indicator specifying which one or more nodes are available as lock owners; and executable code that sends a lock request message to said current lock owner, and wherein said executable code that determines is executed by said requesting node prior to executing said executable code that sends for each request and wherein, each time said lock is granted, an automatic expiration period is determined indicating when a current grant of said lock at said each time automatically expires. Claims 41-43 depend from Claim 40.

Fee relates to mechanisms for resolving address contention and prioritization of access to resources within a shared memory system. (Col. 1, Lines 7-1). Fee discloses creating ordered lists of requests for each contested resource. A new request is added to the lists only after a

conflict is recognized. (See abstract). Fee discloses that as the shared memory controller processes each request, address contention is checked. If there is no current owner designated to access the specified address range, that request is granted ownership of that address space until its request is satisfied. Subsequent requests for the same memory location set their need for the last requester. As each master completes, only one requester sets its need and is processed. (Col. 3, Lines 44-51).

Claim 1, as amended herein, is neither disclosed nor suggested by Fee in that Fee neither discloses nor suggests at least the features of *a method for distributed lock management comprising: determining, by a requesting node requesting a lock, a current lock owner of the lock in accordance with an indicator specifying which one or more nodes are available as lock owners; and sending a lock request message to said current lock owner; and wherein ... said determining is performed by said requesting node prior to said sending for each request, and wherein, each time said lock is granted, an expiration time is determined indicating that said lock expires automatically when said expiration time is reached*, as set forth in Claim 1.

Fee discloses that all requests to the storage controller SC are initiated by sending an address and command to the SC from a CP, I/O adapter, or other remote SC. (See, for example, Col. 6, Line 66-Col. 7, Line 1; Col. 3, Lines 44-57). Fee discloses CP1 (central processor 1) receiving its data and its requester in the SC completes resetting its valid latch. Fee discloses that CP2 has its need register set for CP1 and resets its need register allowing CP2 to make a new request to the pipeline. (See Col. 10, Line 64-Col. 11, Line 1). Fee discloses sending requests to the storage controller and resetting need registers of CPs. However, Fee neither discloses nor

suggests sending a lock request message to the current lock owner. Thus, Fee neither discloses nor suggests *sending a lock request message to said current lock owner*, as set forth in Claim 1.

Fee discloses that the shared memory controller determines whether there is a current owner (e.g., address contention) for a specified address range after a request is received by the controller. (See, for example, Col. 3, Lines 44-57). However, Fee neither discloses nor suggests that a requesting node makes such a determination. Furthermore, Fee discloses that a determination regarding current ownership is made after a request is received by the controller not prior to sending the request. Thus, Fee neither discloses nor suggests *determining, by a requesting node requesting a lock, a current lock owner of the lock in accordance with an indicator specifying which one or more nodes are available as lock owners;... wherein ... said determining is performed by said requesting node prior to said sending for each request*, as set forth in Claim 1.

Fee discloses four processors attempting to access a common address and discloses that the four processors enter the pipeline in their named order P0, P1, P2, and then P3 (see Col. 6, Lines 19-36). However, Fee appears silent regarding any mention of determining an expiration time and neither discloses nor suggests *wherein, each time said lock is granted, an expiration time is determined indicating that said lock expires automatically when said expiration time is reached*, as set forth in Claim 1.

For at least these reasons, Applicant's Claim 1, and claims that depend therefrom, are neither disclosed nor suggested by Fee.

Independent Claims 18, 22, 23, and 40 recite features similar to those set forth above regarding Claim 1 which are neither disclosed nor suggested by Fee. Thus, Claims 18, 22, 23, and 40, and claims that depend therefrom, are neither disclosed nor suggested by Fee for reasons similar to those set forth above regarding Claim 1.

In view of the foregoing, Applicant respectfully requests that the rejection be reconsidered and withdrawn.

The rejection of Claims 1-43 under 35 U.S.C. 102(e) as being unpatentable over Jeffords et al. (U.S. Patent No. 6,510,478, hereinafter referred to as “Jeffords”) is hereby traversed and reconsideration thereof is respectfully requested.

Claims 1-43 are summarized above.

Jeffords is directed to controlling access to an object shared amongst a plurality of processes in a distributed system so that the shared object is accessed by one and only one process at a time. (Col. 1, Lines 27-30). Jeffords discloses that one process is identified as a lock owner process controlling a lock associated with the shared object. When no other process controls the lock, the lock owner process grants control to the requesting process. When another process controls the lock, the lock owner process places the requesting process in a queue and waits for the lock to become available. All accesses to the shared object are processed through the lock owner processes thus assuring coordination and synchronization among the processes. (see Abstract). Jeffords discloses that once a requesting process acquires a lock, the lock is held until it is returned to the lock owner. (Col. 4, Lines 32-34). Jeffords also discloses that once a

requesting process acquires the lock, the lock owner process will not give the lock to any other process that requests it until the lock holder processes either releases the lock or the lock holder process fails. (Col. 5, Lines 14-18). Jeffords discloses that a lock is released by sending a release lock message to the lock owner process (See, for example, Col. 4, Lines 62-65; Figure 3 element 316).

Claim 1, as amended herein, is neither disclosed nor suggested by Jeffords in that Jeffords neither discloses nor suggests at least the features of *a method for distributed lock management comprising: ... wherein, each time said lock is granted, an expiration time is determined indicating that said lock expires automatically when said expiration time is reached*, as set forth in Claim 1.

As described above, Jeffords discloses that once a requesting process acquires the lock, the lock owner process will not give the lock to any other process that requests it until the lock holder processes either releases the lock or the lock holder process fails. Jeffords discloses that a lock is released by sending a release lock message to the lock owner process. However, Jeffords neither discloses nor suggests *wherein, each time said lock is granted, an expiration time is determined indicating that said lock expires automatically when said expiration time is reached*, as set forth in Claim 1.

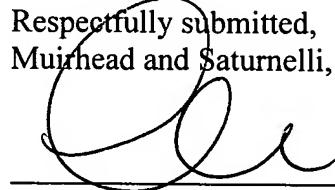
For at least these reasons, Applicant's Claim 1, and claims that depend therefrom, are neither disclosed nor suggested by Jeffords.

Independent Claims 18, 22, 23, and 40 recite features similar to those set forth above regarding Claim 1 which are neither disclosed nor suggested by Jeffords. Thus, Claims 18, 22, 23, and 40, and claims that depend therefrom, are neither disclosed nor suggested by Jeffords for reasons similar to those set forth above regarding Claim 1.

In view of the foregoing, Applicant respectfully requests that the rejection be reconsidered and withdrawn.

Based on the above, Applicant respectfully requests that the Examiner reconsider and withdraw all outstanding rejections and objections. Favorable consideration and allowance are earnestly solicited. Should there be any questions after reviewing this paper, the Examiner is invited to contact the undersigned at 508-898-8604.

Respectfully submitted,  
Muirhead and Saturnelli, LLC



---

Anne E. Saturnelli  
Reg. No. 41,290

Muirhead and Saturnelli, LLC  
200 Friberg Parkway, Suite 1001  
Westborough, MA 01581  
Tel: (508) 898-8601  
Fax: (508) 898-8602

Date: August 9, 2007